

3D Face Reconstruction with Geometry Details from a Single Image

Luo Jiang, Juyong Zhang, Bailin Deng, *Member, IEEE*, Hao Li, and Ligang Liu

Abstract—3D face reconstruction from a single image is a classical and challenging problem, with wide applications in many areas. Inspired by recent works in face animation from RGB-D or monocular video inputs, we develop a novel method for reconstructing 3D faces from unconstrained 2D images, using a coarse-to-fine optimization strategy. First, a smooth coarse 3D face is generated from an example-based bilinear face model, by aligning the projection of 3D face landmarks with 2D landmarks detected from the input image. Afterwards, using global corrective deformation fields, the coarse 3D face is refined using photometric consistency constraints, resulting in a medium face shape. Finally, a shape-from-shading method is applied on the medium face to recover fine geometric details. Our method outperforms state-of-the-art approaches in terms of accuracy and detail recovery, which is demonstrated in extensive experiments using real world models and publicly available datasets.

Index Terms—Tensor Model, Shape-from-shading, 3D Face Reconstruction.

I. INTRODUCTION

Reconstruction of 3D face models using 2D images is a fundamental problem in computer vision and graphics [1], with various applications such as face recognition [2], [3] and animation [4], [5]. However, this problem is particularly challenging, due to the loss of information during camera projection.

In the past, a number of methods have been proposed for face construction using a single image. Among them, example-based methods first build a low-dimensional parametric representation of 3D face models from an example set, and then fit the parametric model to the input 2D image. One of the most well-known example is the 3D Morphable Model (3DMM) proposed by Blanz and Vetter [6], represented as linear combination of the example faces. 3DMM is a popular parametric face model due to its simplicity, and has been the foundation of other more sophisticated face reconstruction methods [3]. Another approach to single image reconstruction is to solve it as *Shape-from-shading* (SFS) [7], a classical computer vision problem of 3D shape recovery from shading variation. For example, Kemelmacher-Shlizerman and Basri [8] reconstruct the depth information from an input face image, by estimating its lighting and reflectance parameters using a reference face shape.

While these existing approaches are able to produce high-quality reconstruction from a single image, they also come with limitations. Although example-based methods are simple

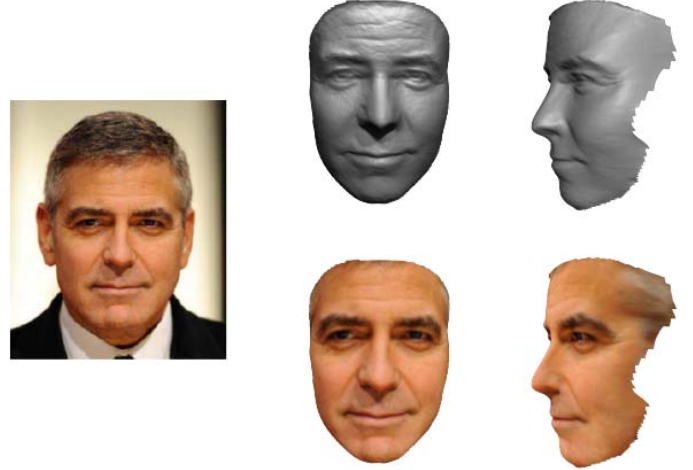


Figure 1: 3D face reconstruction from a single image. Given an input image (left), we reconstruct a 3D face with fine geometric details (right, top row). The input image can be used as texture for rendering the reconstructed face (right, bottom row).

and efficient, they rely heavily on the dataset, and may produce unsatisfactory results when the target face is largely different from those in the example set; moreover, due to the limited degrees of freedom of the low-dimensional model, these methods often fail to reproduce fine geometric details (such as wrinkles) that are specific to the target face. SFS-based methods are able to capture the fine-scale facial details from the appearance of the input image; however, they require prior knowledge about the geometry or illumination to resolve the ambiguity of the reconstruction problem, and may become inaccurate when the input image does not satisfy the assumptions.

In this paper, we propose a novel coarse-to-fine method to reconstruct a high-quality 3D face model from a single image. Our method consists of three steps:

- First, we compute a coarse estimation of the target 3D face, by fitting an example-based parametric face model to the input image. Our parametric model is derived from FACEWAREHOUSE [9], a 3D face dataset with large variation in identity and expression. The resulting mesh model captures the overall shape of the target face.
- Afterwards, we enhance the coarse face model by applying smooth deformation that captures medium-scale facial features; we also estimate the lighting and reflectance parameters from the enhanced face model.
- Finally, the illumination parameters and the enhanced face model are utilized to compute a height-field face surface

L. Jiang, J. Zhang, H. Li, L. Liu are with School of Mathematical Sciences, University of Science and Technology of China. E-mail: jl原因@mail.ustc.edu.cn, juyong@ustc.edu.cn, lihao215@mail.ustc.edu.cn, lgliu@ustc.edu.cn.

B. Deng is with School of Computer Science and Informatics, Cardiff University. E-mail: DengB3@cardiff.ac.uk.

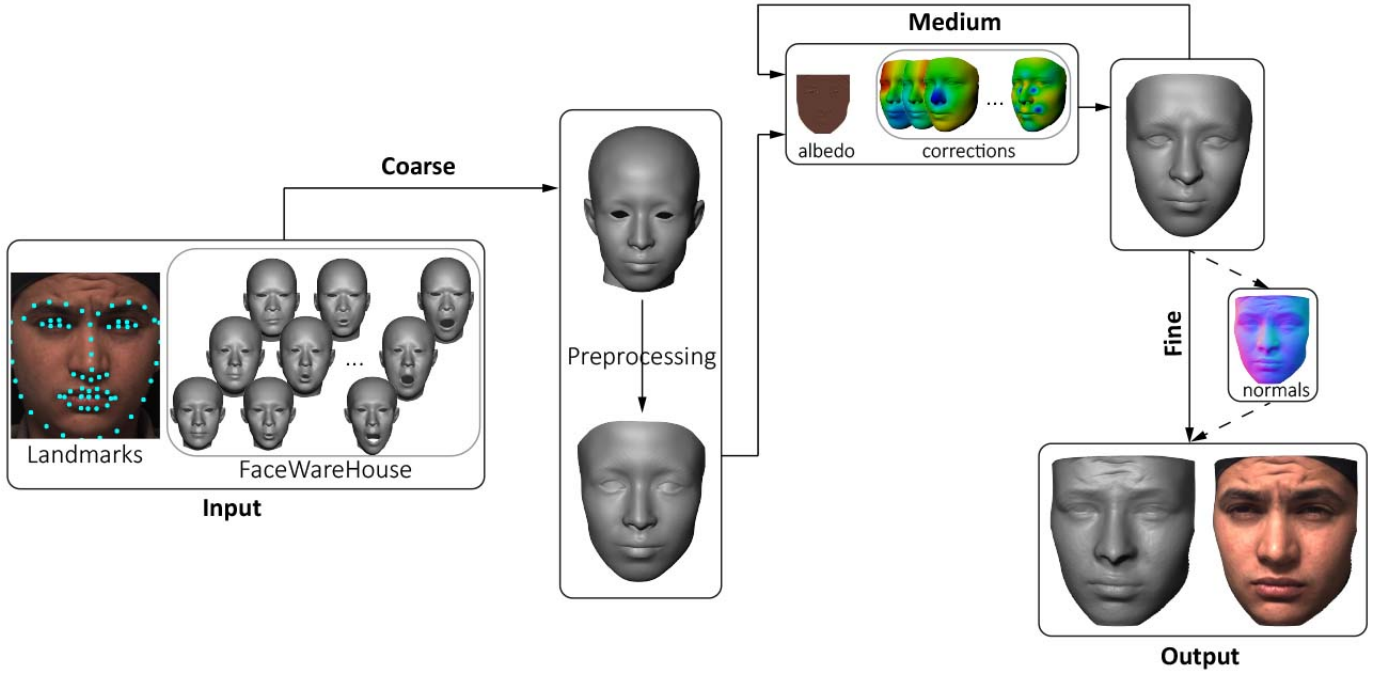


Figure 2: An overview of our coarse-to-fine face reconstruction approach.

according to the shading variation of the input image. This final model faithfully captures the fine geometric details of the target face (see Fig. 1).

Our method builds upon the strength of the existing approaches mentioned above: the example-based coarse face enables more reliable estimation of illumination parameters, and improves the robustness of the final SFS step; the SFS-based final face model provides detailed geometric features, which are often not available from example-based approaches. Our method outperforms existing example-based and SFS methods in terms of reconstruction accuracy as well as geometric detail recovery, as shown by extensive experimental results using publicly available datasets.

II. RELATED WORK

Low-dimensional models. Human faces have similar global characteristics, for example the location of main facial features such as eyes, nose and mouth. From a perception perspective, it has been shown that a face can be characterized using a limited number of parameters [10], [11]. The low dimensionality of the face space allows for effective parametric face representations that are derived from a collection of sample faces, reducing the reconstruction problem into searching within the parameter space. A well-known example of such representations is the 3DMM proposed in [6], which has been used for various face processing tasks such as reconstruction [6], recognition [2], [3], face exchange in images [12], and makeup suggestion [13]. Low-dimensional representations have also been used for dynamic face processing. To transfer facial performance between individuals in different videos, Vlasic et al. [14] develop a multilinear face model representation that separately parameterizes different face attributes such as identity, expression, and viseme. In the computer graphics

industry, facial animation is often achieved using linear models called blendshapes, where individual facial expressions are combined to create realistic facial movements [15]. The simplicity and efficiency of blendshapes models enable real-time facial animation driven by facial performance captured from RGBD cameras [16], [17], [18], [19], [20] and monocular videos [21], [4], [22], [5]. When using low-dimensional face representations derived from example face shapes, the example dataset has strong influence on the resulting face models. For instance, it would be difficult to reconstruct a facial expression that deviates significantly from the sample facial expressions. In the past, during the development of face recognition algorithms, various face databases have been collected and made publicly available [23]. Recently, Cao et al. [9] introduce FACEWAREHOUSE, a 3D facial expression database that provides the facial geometry and texture of 150 subjects, covering a wide range of ages and ethnic backgrounds. Our coarse face modeling method adopts a bilinear face model that encodes identity and expression attributes in a way similar to [14]. We use FACEWAREHOUSE as the example dataset, due to the variety of facial expressions and identities that it provides.

Shape-from-shading. Shape-from-shading (SFS) [7], [24] is a computer vision technique that recovers 3D shapes from their shading variation in 2D images. Given the information about illumination, camera projection, and surface reflectance, SFS methods are able to recover fine geometric details that may not be available using low-dimensional models. On the other hand, SFS is an ill-posed problem with potentially ambiguous solutions [25]. Thus for face reconstruction, prior knowledge about facial geometry must be incorporated to achieve reliable results. For example, symmetry of human faces has been used by various authors to reduce the ambiguity

of SFS results [26], [27], [28]. Another approach is to solve the SFS problem within a human face space, using a low-dimensional face representation [29], [30]. Other approaches improve the robustness of SFS by introducing extra data source, such as a separate reference face [8], as well as coarse reconstructions using multiview stereo [31], [32] or unconstrained photo collections [33], [34], [35]. We adopt a similar approach which builds an initial estimation of the face shape and augment it with fine geometric details using SFS. Our initial face estimation combines coarse reconstruction in a low-dimensional face space with refinement of medium-scale geometric features, providing a more accurate initial shape for subsequent SFS processing.

III. OVERVIEW

This section provides an overview of our coarse-to-fine approach to reconstructing a high-quality 3D face model from a single photograph. Fig. 2 illustrates the pipeline of our method.

To create a coarse face model (Sec. IV), we first build a bilinear model from FACEWAREHOUSE to describe a plausible space of 3D faces; the coarse face shape is generated from the bilinear model by aligning the projection of its 3D landmarks with the 2D landmarks detected on the input image, using a fitting energy that jointly optimizes the shape parameters (e.g., identity, expression) and camera parameters. To further capture person-specific features that are not available from the bilinear model, we enhance the coarse face using an additional deformation field that corresponds to medium-scale geometric features (Sec. V); the deformation field is jointly optimized with the lighting and albedo parameters, such that the shading of the enhanced model is close to the input image. Afterwards, the resulting medium face model is augmented with fine geometric details (Sec. VI): the normal field from the medium face model is modified according to the input image gradients as well as the illumination parameters derived previously, and the modified normal field is integrated to achieve the final face shape.

IV. COARSE FACE MODELING

Bilinear face model. The FACEWAREHOUSE data set contains facial expressions of 150 individuals, each with 47 expressions. All expressions are represented as meshes of the same connectivity, each consisting of 11510 vertices. Following [14], we collect the vertex coordinates of all meshes into a third-order data tensor, and perform 2-mode SVD reduction along the identity mode and the expression mode, to derive a bilinear face model that approximates the original data set. In detail, the bilinear face model is represented as a mesh with the same connectivity as those from the data set, and its vertex coordinates $\mathbf{F} \in \mathbb{R}^{3 \times 11510}$ are computed as

$$\mathbf{F} = \mathbf{C}_r \times_2 \mathbf{w}_{\text{id}}^T \times_3 \mathbf{w}_{\text{exp}}^T, \quad (1)$$

where \mathbf{C}_r is the reduced core tensor computed from the SVD reduction, and $\mathbf{w}_{\text{id}} \in \mathbb{R}^{50}$, $\mathbf{w}_{\text{exp}} \in \mathbb{R}^{47}$ are column vectors for the identity weights and expression weights which control the face shape. Note that here we only reduce the dimension along the identity mode, in order to maintain the variety of

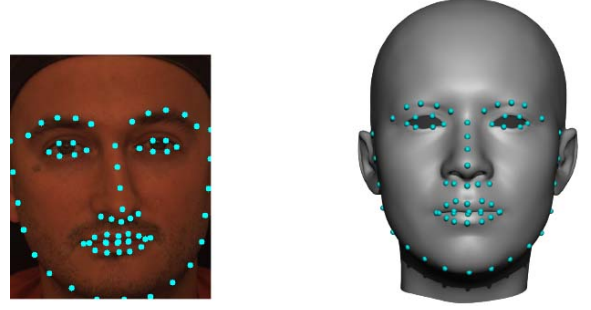


Figure 3: Our coarse face reconstruction is based on aligning the projection of labeled 3D face landmarks (right) with 2D landmarks detected on the input image (left).

facial expressions in the bilinear model. For more details on multilinear algebra, the reader is referred to [36].

To construct a coarse face, we align 3D landmarks on the bilinear face model with corresponding 2D landmarks from the input image. First, we preprocess the bilinear face mesh to manually label 68 landmark vertices. Given an input image, we detect the face as well as its corresponding 68 landmarks using the method in [37] (see Fig. 3 for an example). Assuming that the camera model is a weak perspective projection along the Z direction, we can write the projection matrix as $\mathbf{\Pi} = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \alpha & 0 \end{bmatrix}$. Then we can formulate the following fitting energy to align the projection of landmark vertices with the detected 2D landmarks

$$E_{\text{fit}} = \sum_{k=1}^{68} \|\mathbf{\Pi R F}_{v_k} + \mathbf{t} - \mathbf{U}_k\|_2^2 + \gamma_1 \sum_{i=1}^{50} \left(\frac{w_{\text{id}}^{(i)}}{\delta_{\text{id}}^{(i)}} \right)^2 + \gamma_2 \sum_{j=1}^{47} \left(\frac{w_{\text{exp}}^{(j)}}{\delta_{\text{exp}}^{(j)}} \right)^2. \quad (2)$$

Here $\mathbf{F}_{v_k} \in \mathbb{R}^3$ and $\mathbf{U}_k \in \mathbb{R}^2$ are the coordinates of the k -th 3D landmark vertex and the corresponding image landmark, respectively; translation vector $\mathbf{t} \in \mathbb{R}^3$ and rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ determine the position and pose of the face mesh with respect to the camera; $w_{\text{id}}^{(i)}$ and $w_{\text{exp}}^{(j)}$ are components of weight vectors \mathbf{w}_{id} and \mathbf{w}_{exp} , while $\delta_{\text{id}}^{(i)}$ and $\delta_{\text{exp}}^{(j)}$ are the corresponding singular values obtained from the 2-mode SVD reduction; γ_1 and γ_2 are positive weights. As in [6], the last two terms ensure parameters $w_{\text{id}}^{(i)}$ and $w_{\text{exp}}^{(j)}$ have a reasonable range of variation. This fitting energy is minimized with respect to the shape parameters \mathbf{w}_{id} , \mathbf{w}_{exp} and the camera parameters $\mathbf{\Pi}$, \mathbf{R} , \mathbf{t} via coordinate descent. First we fix the shape parameters and reduce the optimization problem to

$$\min_{\mathbf{\Pi}, \mathbf{R}, \mathbf{t}} \sum_{k=1}^{68} \|\mathbf{\Pi R F}_{v_k} + \mathbf{t} - \mathbf{U}_k\|_2^2, \quad (3)$$

which is solved using the pose normalization method from [33]. Next we fix the camera and expression parameters, which turns the optimization into

$$\min_{\mathbf{w}_{\text{id}}} \sum_{k=1}^{68} \|\mathbf{\Pi R F}_{v_k} + \mathbf{t} - \mathbf{U}_k\|_2^2 + \gamma_1 \sum_{i=1}^{50} \left(\frac{w_{\text{id}}^{(i)}}{\delta_{\text{id}}^{(i)}} \right)^2. \quad (4)$$

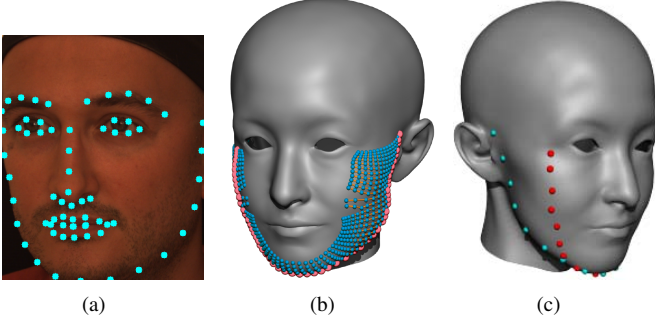


Figure 4: For a non-frontal face images (a), the labeled 3D face silhouette landmarks (shown in cyan in (c)) need to be updated for better correspondence with the detected 2D contour landmarks. We construct a set of horizontal lines connecting the mesh vertices (shown in blue in (b)), and select among them a set of vertices representing the updated silhouette according to the current view direction (shown in pink in (b)). The new 3D silhouette landmarks (shown in red in (c)) are selected within the updated silhouette.

This is a linear least-squares problem and can be solved easily. Finally, we fix the camera and identity parameters, and optimize the expression parameters in the same way as (4). We iteratively run these steps until convergence. In our experiments, four iterations are sufficient for convergence to a good result.

Landmark vertex update. The landmark vertices on the face mesh are labeled based on the frontal pose. For non-frontal face images, the detected 2D landmarks along the face contour may not correspond well with the landmark vertices (see Fig. 4(a) for an example). Thus after each camera parameter optimization step, we update the silhouette landmark vertices according to the rotation matrix \mathbf{R} , while keeping the internal landmark vertices (e.g., those around the eyes, the nose, and the mouth) unchanged. Similar to [4], we preprocess the original face mesh to derive a dense set of horizontal lines that connect mesh vertices and cover the potential silhouette region from a rotated view (see Fig. 4(b)). Given a rotation matrix \mathbf{R} , we select from each horizontal line a vertex that lies on the silhouette, and project it onto the image plane according to the camera parameters $\mathbf{\Pi}, \mathbf{R}, \mathbf{t}$. These projected vertices provide an estimate of the silhouette for the projected face mesh. Then for each 2D contour landmark, its corresponding landmark vertex is updated to the silhouette vertex whose projection is closest to it (see Fig. 4(c)).

To determine the silhouette vertex on a horizontal line, we select the vertex whose normal encloses the largest angle with the view direction. Since the face mesh is approximately spherical with its center close to the origin, we approximate the unit normal of a vertex on the rotated face mesh as $\frac{\mathbf{R}\mathbf{v}}{\|\mathbf{R}\mathbf{v}\|_2}$, where \mathbf{v} is the original vertex coordinates. Then the silhouette vertex is the one with the smallest value of $\left| \mathbf{Z} \cdot \frac{\mathbf{R}\mathbf{v}}{\|\mathbf{R}\mathbf{v}\|_2} \right|$ within the horizontal line, where $\mathbf{Z} = [0, 0, 1]^T$ is the view direction.

V. MEDIUM FACE MODELING

Although the coarse face model provides a good estimate of the overall shape, it may not capture some person-specific geometric details due to limited variation of the FACEWAREHOUSE data set (see Fig. 6). Thus we enhance the coarse face using smooth deformation that correspond to medium-scale geometric features, to improve the consistency between its shading and the input image. During this process we also estimate the lighting and the albedo. The enhanced face model and the lighting/albedo information will provide the prior knowledge required by the SFS reconstruction in the next section.

Preprocessing. The coarse face model computed in Sec. IV is a full head mesh, aligned with the camera using the optimized rotation \mathbf{R} and translation \mathbf{t} . It has about 5K vertices on the frontal part, and has holes on the eyes. To allow for recovery of more geometric details, we preprocess the bilinear face mesh to identify the vertices that belong to the frontal part, and use this information to extract the frontal region of the coarse face as a mesh; we fill the holes on the extracted mesh, and apply two levels of Loop subdivision [38] to increase its resolution to about 70K vertices. This frontal face mesh will be used for the following processing. In this paper, we convert color input images into grayscale ones, as the formulation below assumes grayscale images for simplicity and efficiency. However, it is not difficult to generalize this formulation to images with multiple color channels, to avoid conversion to grayscale images.

Lighting and albedo estimation. To compute shading for our face mesh, we need the information about lighting and surface reflectance. Assuming Lambertian reflectance, we can approximate the grayscale level $s_{i,j}$ at a pixel (i, j) using second-order spherical harmonics [39]:

$$s_{i,j} = r_{i,j} \cdot \max(\boldsymbol{\xi}^T \mathbf{H}(\mathbf{n}_{i,j}), 0). \quad (5)$$

Here $r_{i,j}$ is the albedo at the pixel; $\mathbf{n}_{i,j}$ is the corresponding mesh normal, computed via

$$\mathbf{n}_{i,j} = \frac{(\mathbf{v}_2^{i,j} - \mathbf{v}_1^{i,j}) \times (\mathbf{v}_3^{i,j} - \mathbf{v}_1^{i,j})}{\|(\mathbf{v}_2^{i,j} - \mathbf{v}_1^{i,j}) \times (\mathbf{v}_3^{i,j} - \mathbf{v}_1^{i,j})\|_2}, \quad (6)$$

where $\mathbf{v}_1^{i,j}, \mathbf{v}_2^{i,j}, \mathbf{v}_3^{i,j}$ are the vertex coordinates for the mesh triangle that corresponds to pixel (i, j) ; \mathbf{H} is a vector of second-order spherical harmonics

$$\mathbf{H}(\mathbf{n}) = [1, n_x, n_y, n_z, n_x n_y, n_x n_z, n_y n_z, n_x^2 - n_y^2, 3n_z^2 - 1]^T, \quad (7)$$

and $\boldsymbol{\xi}$ is a vector of harmonics coefficients. Although such approximation works well for a large variety of shapes and lighting conditions, it does not account for more complex settings such as cast shadows, which can affect the accuracy of the light/albedo estimate. For more robust estimation, we follow [32] and introduce a smooth corrective scalar field to compensate for the inaccuracy of the illumination model:

$$s_{i,j} = r_{i,j}(\boldsymbol{\xi}^T \mathbf{H}(\mathbf{n}_{i,j}) + d_{i,j}), \quad (8)$$

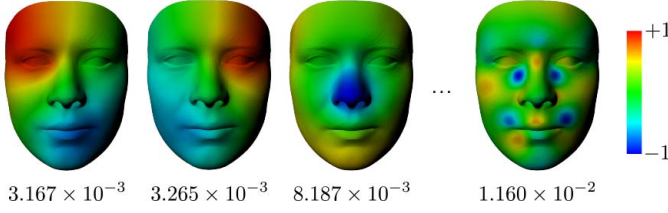


Figure 5: Some Laplacian eigenfunctions of the face mesh (displayed via color coding) and their eigenvalues. Eigenfunctions with larger eigenvalues oscillate more rapidly.

where $d_{i,j}$ is the corrective value at the pixel. The lighting and albedo are then estimated by solving an optimization problem

$$\min_{\mathbf{r}, \boldsymbol{\xi}, \mathbf{d}} \sum_{i,j} (r_{i,j} (\boldsymbol{\xi}^T \mathbf{H}(\mathbf{n}_{i,j}) + d_{i,j}) - I_{i,j})^2 + \mu_1 \|\mathbf{d}\|_2^2 + \mu_2 \|\mathbf{G}\mathbf{d}\|_2^2 + \mu_3 \|\mathbf{L}\mathbf{d}\|_2^2, \quad (9)$$

where vectors \mathbf{r} , \mathbf{d} collect the values $\{r_{i,j}\}$, $\{d_{i,j}\}$, respectively; $I_{i,j}$ denotes the grayscale value at pixel (i,j) of the input image; \mathbf{G} , \mathbf{L} are the gradient and Laplacian matrices, respectively; μ_1, μ_2, μ_3 are user-specified positive weights specified by the user. The last three terms in (9) regularizes the scalar field \mathbf{d} and ensures low frequency of illumination [32]. To optimize this problem, we first set all $\{r_{i,j}\}$ to be the median grayscale value of the input face pixels, fix \mathbf{d} to zero, and optimize the harmonics coefficients $\boldsymbol{\xi}$. Then we optimize the corrective field \mathbf{d} while fixing $\boldsymbol{\xi}$ and \mathbf{r} . Both sub-problems reduce to solving a linear system. Finally, we compute the albedo as $r_{i,j} = I_{i,j} / (\boldsymbol{\xi}^T \mathbf{H}(\mathbf{n}_{i,j}) + d_{i,j})$.

Facial detail enhancement. With an estimate of lighting and albedo, we can now enhance the coarse face mesh to reduce the discrepancy between the mesh shading and the input image. We apply a smooth 3D deformation field to the n vertices of the frontal face mesh to minimize the following discrepancy measure with respect to the vertex displacements $\mathbf{D} \in \mathbb{R}^{3 \times n}$:

$$E_{\text{shading}}(\mathbf{D}) = \sum_{i,j} (r_{i,j} \max(\boldsymbol{\xi}^T \mathbf{H}(\tilde{\mathbf{n}}_{i,j}), 0) - I_{i,j})^2, \quad (10)$$

where $\{\tilde{\mathbf{n}}_{i,j}\}$ are the new mesh face normals. However, this nonlinear least-squares problem can be very time-consuming to solve, due to the high resolution of the mesh. Therefore, we construct a low-dimensional subspace of smooth mesh deformations and solve the optimization problem within this subspace, which significantly reduces the number of variables. Specifically, if we measure the smoothness of a deformation field using the norm of its graph Laplacian with respect to the mesh, then the Laplacian eigenfunctions associated with small eigenvalues span a subspace of smooth deformations. Indeed, it is well known in 3D geometry processing that the Laplacian eigenvalues can be seen as the frequencies for the eigenfunctions, which indicate how rapidly each eigenfunction oscillates across the surface [40] (see Fig. 5). Thus by restricting the deformation to the subspace with small eigenvalues, we inhibit the enhancement of fine-scale geometric features, leaving them to the SFS reconstruction step in Sec VI.

To perform subspace optimization, we preprocess the frontal face mesh to construct its graph Laplacian matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$

based on mesh connectivity, and apply eigendecomposition to obtain $k+1$ eigenvectors $\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_k$ corresponding to the smallest eigenvalues $\lambda_0 \leq \lambda_1 \leq \dots \leq \lambda_k$. Among them, \mathbf{e}_0 has the same value for all its components, and represents translation of the whole mesh [40]. Since it does not alter the facial geometry, we discard \mathbf{e}_0 from our subspace of deformations. Using the remaining eigenvectors to span the x -, y -, and z -coordinates of the vertex displacement vectors, we can represent the deformation field \mathbf{D} via

$$\mathbf{D} = (\mathbf{E}\boldsymbol{\eta})^T, \quad (11)$$

where $\mathbf{E} = [\mathbf{e}_1, \dots, \mathbf{e}_k] \in \mathbb{R}^{n \times k}$ stacks the basis vectors, and $\boldsymbol{\eta} = [\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_k]^T \in \mathbb{R}^{k \times 3}$ collects the coefficients for the basis vectors. Then the deformation is computed by solving the following optimization problem about $\boldsymbol{\eta}$:

$$\min_{\boldsymbol{\eta}} E_{\text{shading}}(\mathbf{D}) + \mu_4 \sum_{i=1}^k \left\| \frac{\boldsymbol{\eta}_i}{\lambda_i} \right\|_2^2. \quad (12)$$

Here the second term prevents large deformations, with more penalty on basis vectors of lower frequencies; μ_4 is a user-specified weight. Our formulation is designed to induce more enhancement for finer geometric features, since the coarse face already provides a good estimate of the overall shape. In our experiments, 40 basis vectors are sufficient for good results.

As the number of variables are significantly reduced in (12), this nonlinear least-squares problem can be solved efficiently using the Levenberg-Marquardt algorithm [41]. We then apply the optimized deformation field to the frontal face mesh, and update the correspondence between image pixels and mesh triangles. With the new correspondences, we solve the optimization problems (9) and (12) again to further improve the lighting/albedo estimate and the face model. This process is iterated twice in our experiments. Fig. 6 shows the enhanced face model after each iteration (Figs. 6(b) and 6(c)) for a coarse face (Fig. 6(a)). We can see that the deformation fields help to reduce the reconstruction error.

VI. FINE FACE MODELING

As the final step in our pipeline, we reconstruct a face model with fine geometric details, represented as a height field surface over the face region Ω of the input image. Using the medium face model and the lighting/albedo information computed in Sec. V, we first compute a refined normal map over Ω , to capture the details from the input image. This normal map is then integrated to recover a height field surface for the final face shape.

Overall approach. Specifically, the normal map is defined using a unit vector $\mathbf{n}'_{i,j} \in \mathbb{R}^3$ for each pixel $(i,j) \in \Omega$. Noting that each face pixel corresponds to a normal vector facing towards the camera [8], we represent $\mathbf{n}'_{i,j}$ using two variables $p_{i,j}, q_{i,j}$ as

$$\mathbf{n}'_{i,j} = \frac{(p_{i,j}, q_{i,j}, -1)}{\sqrt{p_{i,j}^2 + q_{i,j}^2 + 1}}. \quad (13)$$

The values $\{p_{i,j}\}$, $\{q_{i,j}\}$ are computed by solving an optimization problem that will be explained later. The final height-field face model, represented using a depth value $z_{i,j}$ per

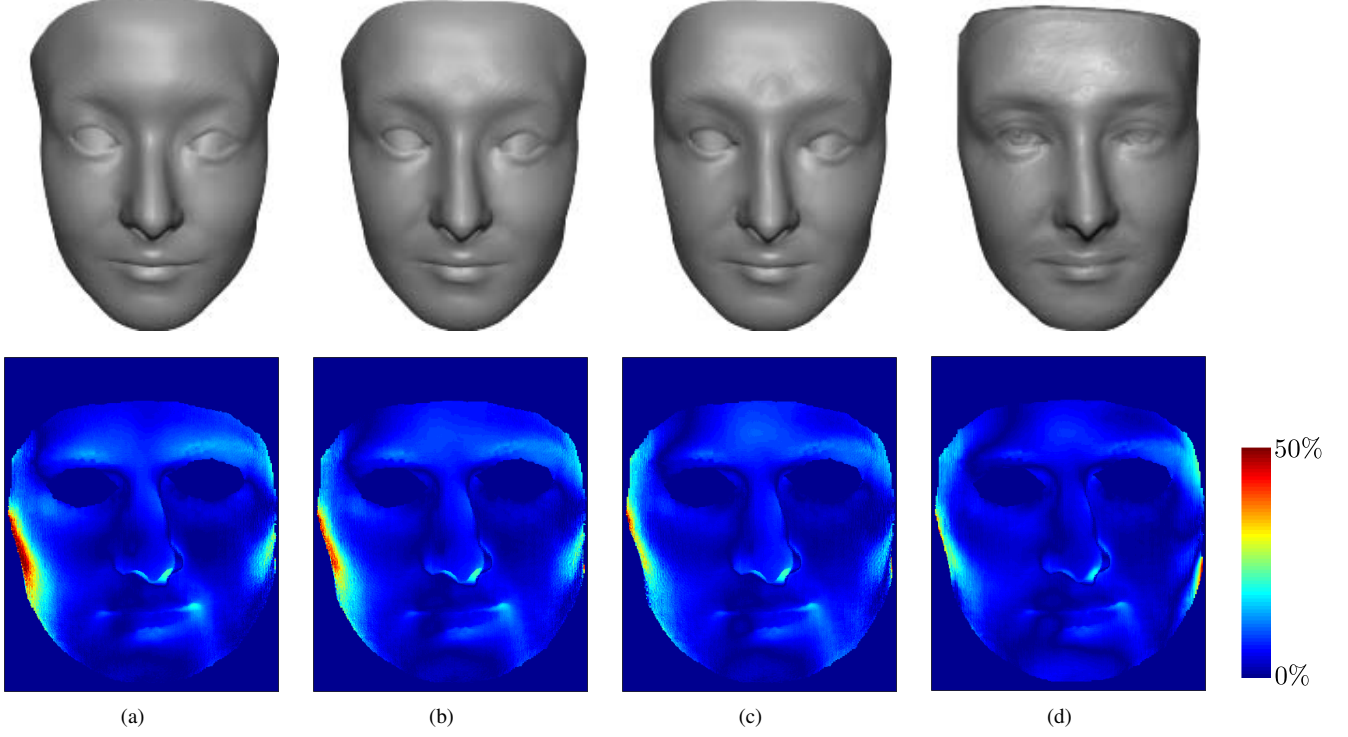


Figure 6: Face models computed at different stages of our pipeline (top row), and their error maps according to Eq. (24) (bottom row). (a): the coarse face shape; (b) and (c): the medium face models after one and two iterations of optimization, respectively; (d): the final face model. The medium face model refines the coarse face and reduces the reconstruction error.

pixel, is then determined so that the height field normals are as close as possible to the normal map. We note that the height field normal $\hat{\mathbf{n}}_{i,j}$ at pixel (i,j) can be computed using three points $\mathbf{h}_{i,j} = (i, j, z_{i,j})$, $\mathbf{h}_{i,j+1} = (i, j+1, z_{i,j+1})$, $\mathbf{h}_{i+1,j} = (i+1, j, z_{i+1,j})$ on the height field surface via

$$\begin{aligned}\hat{\mathbf{n}}_{i,j} &= \frac{(\mathbf{h}_{i,j+1} - \mathbf{h}_{i,j}) \times (\mathbf{h}_{i+1,j} - \mathbf{h}_{i,j})}{\|(\mathbf{h}_{i,j+1} - \mathbf{h}_{i,j}) \times (\mathbf{h}_{i+1,j} - \mathbf{h}_{i,j})\|_2} \\ &= \frac{(z_{i+1,j} - z_{i,j}, z_{i,j+1} - z_{i,j}, -1)}{\sqrt{(z_{i+1,j} - z_{i,j})^2 + (z_{i,j+1} - z_{i,j})^2 + 1}}. \quad (14)\end{aligned}$$

Comparing this with Eq. (13) shows that for the height field normal to be consistent with the normal map, we should have

$$z_{i+1,j} - z_{i,j} = p_{i,j}, \quad z_{i,j+1} - z_{i,j} = q_{i,j} \quad (15)$$

for every pixel. As these conditions only determine $\{z_{i,j}\}$ up to an additional constant, we compute $\{z_{i,j}\}$ as the minimum-norm solution to a linear least-squares problem

$$\min_{\{z_{i,j}\}} \sum_{(i,j)} (z_{i+1,j} - z_{i,j} - p_{i,j})^2 + (z_{i,j+1} - z_{i,j} - q_{i,j})^2. \quad (16)$$

Normal map optimization. For high-quality results, we need to ensure the computed normal map $\mathbf{n}'_{i,j}$ achieve certain desirable properties. We do so by minimizing an energy that enforces these properties. First of all, the normal map should capture fine-scale details from the input image. Using the lighting and albedo parameters obtained during the computation of the medium face, we can evaluate the pixel intensity values from the normal map according to Eq. (5), and require them to

be close to the input image. However, such direct approach can suffer from the inaccuracy of spherical harmonics in complex lighting conditions such as cast shadows, which can lead to unsatisfactory results. Instead, we aim at minimizing the difference in intensity gradients, between the input image and the shading from the normal map. This difference can be measured using the following energy

$$E_{\text{grad}} = \sum_{(i,j)} \left\| \begin{bmatrix} s'_{i+1,j} - s'_{i,j} \\ s'_{i,j+1} - s'_{i,j} \end{bmatrix} - \begin{bmatrix} I_{i+1,j} - I_{i,j} \\ I_{i,j+1} - I_{i,j} \end{bmatrix} \right\|_2^2, \quad (17)$$

where $\{I_{i,j}\}$ are intensity values from the input image, and

$$s'_{i,j} = r_{i,j} \cdot \max(\boldsymbol{\xi}^T \mathbf{H}(\mathbf{n}'_{i,j}), 0) \quad (18)$$

are shading intensities for the normal map according to Eq. (5), using the optimized albedo $\{r_{i,j}\}$ and spherical harmonic coefficients $\boldsymbol{\xi}$ from Sec. V. Minimizing the difference in gradients instead of intensities helps to attenuate the influence from illumination noises such as cast shadows, while preserving the features from the input image.

Optimizing E_{grad} alone is not sufficient for good results, since the problem is under-constrained. Thus we introduce two additional regularization terms for the normal map. First we note that the medium face model from Sec. V provides good approximation of the final shape. Thus we introduce the following energy to penalize the deviation between normal map and the normals from the medium face

$$E_{\text{close}} = \sum_{(i,j)} \|\mathbf{n}'_{i,j} - \mathbf{n}_{i,j}\|_2^2, \quad (19)$$

where $\mathbf{n}_{i,j}$ is computed from the medium face mesh according to Eq. (6). In addition, we enforce smoothness of the normal map using an energy that penalizes its gradient

$$E_{\text{smooth}} = \sum_{(i,j)} \|\mathbf{n}'_{i+1,j} - \mathbf{n}'_{i,j}\|_2^2 + \|\mathbf{n}'_{i,j+1} - \mathbf{n}'_{i,j}\|_2^2. \quad (20)$$

Finally, we need to ensure the normal map is *integrable*, i.e., given the normal map there exists a height field surface such that conditions (15) are satisfied. Note that if (15) are satisfied, then $p_{i,j}$ and $q_{i,j}$ are the increments of function z along the grid directions. Moreover, the total increment of z along the close path that connects pixels (i,j) , $(i+1,j)$, $(i+1,j+1)$, $(i,j+1)$ should be zero, which results in the condition

$$p_{i,j} + q_{i+1,j} - p_{i,j+1} - q_{i,j} = 0. \quad (21)$$

For the normal map to be integrable, this condition should be satisfied at each pixel. Indeed, with condition (15) we can interpret p and q as partial derivatives $\frac{\partial z}{\partial u}$, $\frac{\partial z}{\partial v}$ where u, v are the grid directions; then condition (21) corresponds to $\frac{\partial p}{\partial v} = \frac{\partial q}{\partial u}$, which is the condition for (p, q) to be a gradient field. We can then enforce the integrability condition using an energy

$$E_{\text{int}} = \sum_{(i,j)} (p_{i,j} + q_{i+1,j} - p_{i,j+1} - q_{i,j})^2. \quad (22)$$

Combining the above energies, we derive an optimization problem for computing the desirable normal map

$$\min_{\mathbf{p}, \mathbf{q}} E_{\text{grad}} + \omega_1 E_{\text{close}} + \omega_2 E_{\text{smooth}} + \omega_3 E_{\text{int}}, \quad (23)$$

where the optimization variables \mathbf{p}, \mathbf{q} are the values $\{p_{i,j}\}, \{q_{i,j}\}$, and $\omega_1, \omega_2, \omega_3$ are user-specified weights. This nonlinear least-squares problem is again solved using the Levenberg-Marquardt algorithm.

Fig. 6(d) shows a fine face model reconstructed using our method. Compared with the medium face model, it captures more geometric details and reduces the reconstruction error.

VII. EXPERIMENTS

This section presents experimental results, and compare our method with some existing approaches.

Experimental setup. To verify the effectiveness of our method, we tested it using the data set from the Bosphorus database [42]. This database provides structured-light scanned 3D face point clouds for 105 subjects, as well as their corresponding single-view 2D face photographs. For each subject, the database provides point clouds and images for different facial expressions and head poses. We ran our algorithm on the 2D images, and used the corresponding point clouds as ground truth to evaluate the reconstruction error. 55 subjects with low noises in their point clouds were chosen for testing. As the points from the face point clouds are in correspondence with the pixels of the 2D images, the ground truth faces can be represented using depth values over the image plane. For a reconstructed face, we aligned it with the ground truth face along the depth direction, and computed an error map over the image plane via

$$\varepsilon(i, j) = \frac{|\bar{z}(i, j) - z_{\text{gt}}(i, j)|}{z_{\text{gt}}(i, j)} \quad (24)$$

for each pixel (i, j) , where $\bar{z}(i, j)$ is the depth value for the aligned result face, and $z_{\text{gt}}(i, j)$ is the ground truth depth value. From the error map we also computed the mean and standard deviation of its error values.

We implemented our algorithm in C++ and tested it on a PC with an Intel Core i7-4710MQ 2.50 GHz CPU and 7.5 GB RAM. The weights on optimization problems (2), (9), (12), (23) are chosen as follows: $\gamma_1 = \gamma_2 = 1.5 \times 10^3$; $\mu_1 = 1, \mu_2 = \mu_3 = 2$; $\mu_4 = 6$; $\omega_1 = 6, \omega_2 = 10, \omega_3 = 1$. The nonlinear least-squares problems are solved using the CERES solver [43], with all derivatives evaluated using automatic differentiation. To speed up the algorithm, we downsample the high-resolution 2D images from the database to 30% of their original dimensions before running our algorithm. The down-sampled images have about 400×500 pixels, for which the coarse, medium, and fine face construction steps take about 1 second, 2 minutes, and 1 minute respectively using our nonoptimized implementation.

Frontal and neutral faces. We first tested our method on facial images of frontal pose and neutral expression, from 55 subjects in the Bosphorus database. For comparison we also ran the face reconstruction method from [3], which is based on a 3DMM built from the Basel Face Model [44] and FACEWAREHOUSE. Fig. 7 presents the reconstruction results of 8 subjects using our method and [3], and compares them with the ground truth faces. Thanks to the enhancement in the medium face step and the SFS recovery from the fine face step, our results are closer to the ground truth than the 3DMM-based results from [3]. Our approach can not only obtain a more realistic global facial shape, but also accurately capture the person-specific geometric details such as wrinkles.

Fig. 9 shows the mean reconstruction errors for each of the 55 subjects, using our method and using the method of [3]. It can be seen that the mean error from our results are consistently lower than those from the method of [3]. The overall mean and standard deviation of reconstruction error is $6.28 \pm 5.93\%$ for our results, and $9.34 \pm 9.82\%$ for [3].

Other poses and expressions. We also tested our method on other poses and facial expressions. First, for each of the 55 subjects, we applied our method on their images of neutral expression with three types of poses: Yaw $+10^\circ$, $+20^\circ$, and $+30^\circ$. We computed the mean reconstruction error and standard deviation for each pose, and compared them with the results for the frontal pose using two measures: the maximum variation of mean error (MVME), and the maximum variation of standard deviation (MVSD), defined as

$$\text{MVME} = \max_{i=1,2,3} |\text{ME}_i - \text{ME}_0|, \quad (25)$$

$$\text{MVSD} = \max_{i=1,2,3} |\text{SD}_i - \text{SD}_0|, \quad (26)$$

where ME_0, SD_0 denote the mean error and standard deviation for the frontal pose, and $\text{ME}_i, \text{SD}_i (i = 1, 2, 3)$ denote the values for a non-frontal pose. Fig. 10 shows the MVME and MVSD for each of the 55 subjects. We can see that the MVME is less than 5% in most cases. The overall mean of MVME and MVSD are 2.85% and 7.25% respectively, indicating consistent reconstruction accuracy for different poses.

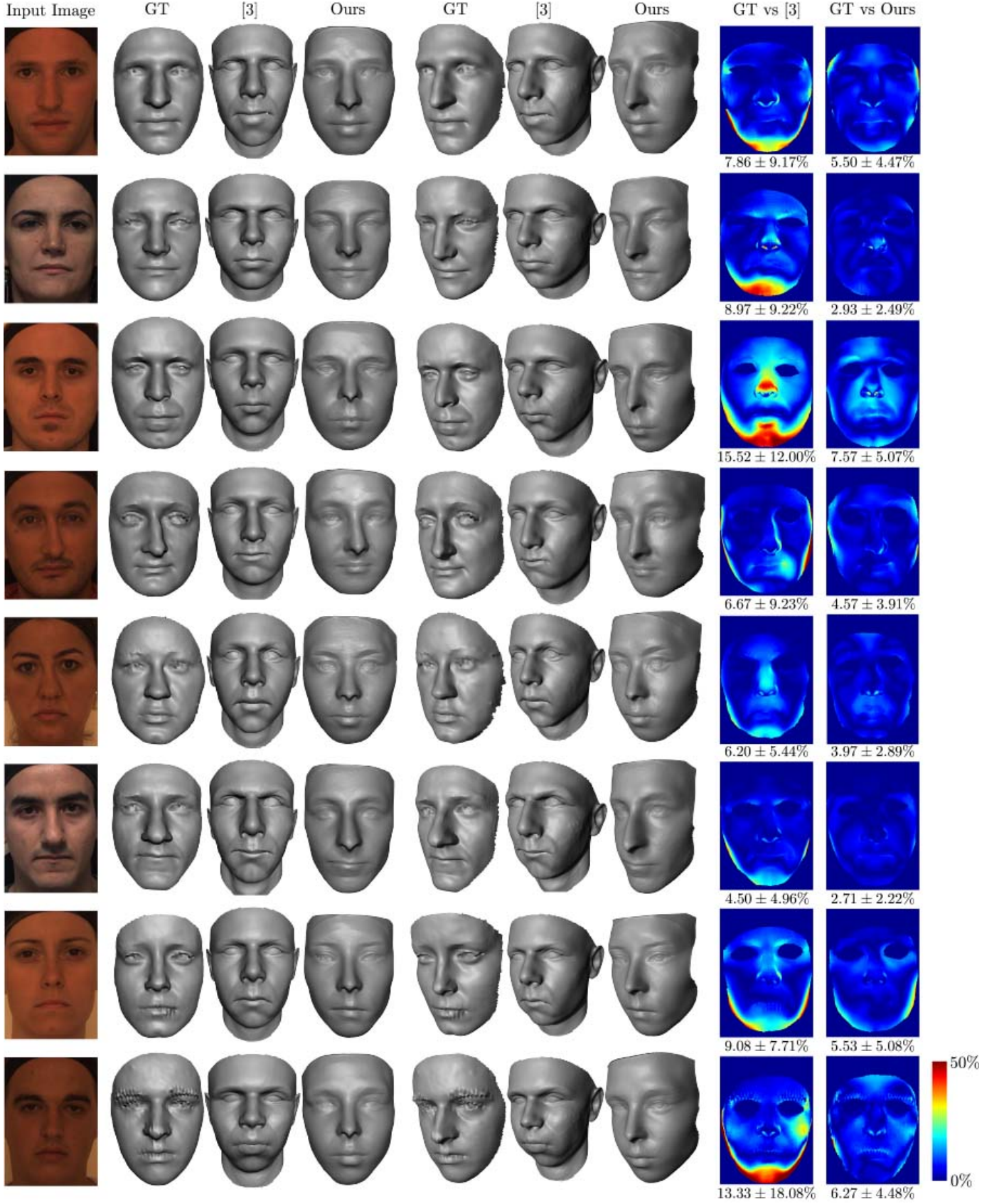


Figure 7: Facial reconstruction from images of frontal pose and neutral expression. For each input image, we show the ground truth (GT) as well as the results using our method and the method from [3], each in two viewpoints. We also show the error maps (according to Eq. (24)) for the two methods, together with their means and standard deviations.



Figure 8: Face reconstructions of four subjects from images of frontal pose with different expressions (happy, surprise, disgust), and of different poses (Yaw $+10^\circ$, $+20^\circ$, $+30^\circ$) with neutral expression. For each input image, we show the reconstructed face mesh as well as its textured rendering.

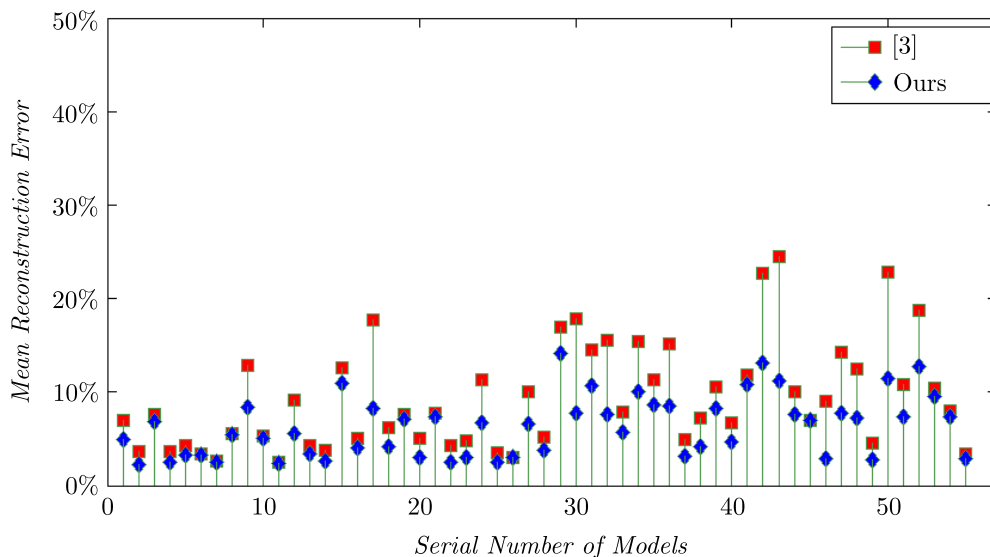


Figure 9: Mean reconstruction errors for frontal-pose, neutral facial images, of 55 subjects from the Bosphorus database. Our method consistently outperformed [3] in terms of mean reconstruction error.

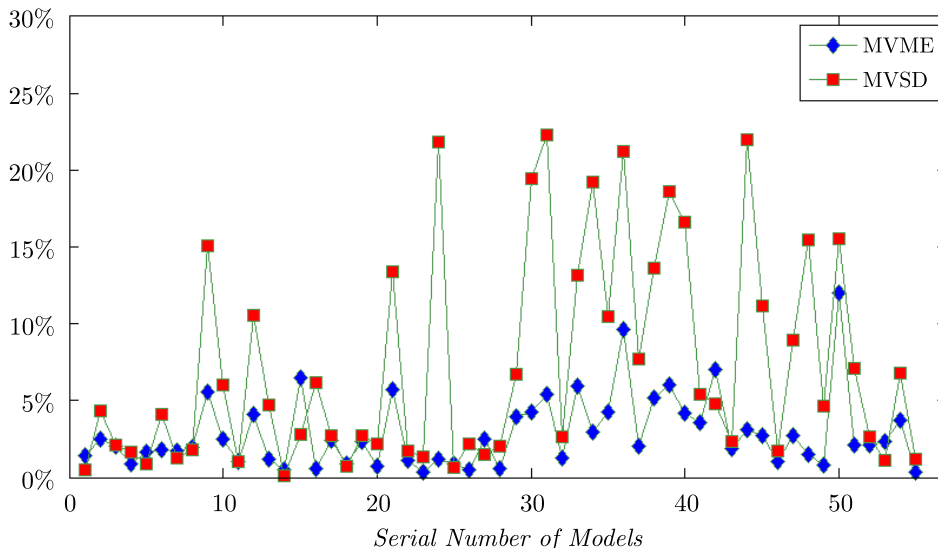


Figure 10: The MVME and MVSD for the reconstruction of 55 subjects from the Bosphorus database, from images with the neutral expression and non-frontal poses (Yaw $+10^\circ$, $+20^\circ$, $+30^\circ$).

Next, we tested our approach on frontal faces with three non-neutral expressions: happy, surprise, and disgust. Among the 55 subjects, there are 25 of them with all three expressions present. We applied our method on these 25 subjects, and compared the reconstruction errors with their neutral expression results using the MVME and MVSD defined in the same way as Eqs. (25) and (26). Fig. 11 shows the MVME and MVSD for each of the 25 subjects. The overall mean of MVME and MVSD are 4.06% and 4.76% respectively, which indicates consistent reconstruction accuracy for different expressions.

In summary, our method achieve consistently accurate results for varying poses and expressions. This can be seen in Fig. 8, where we show the reconstruction results of 4 subjects under different poses and expressions.

Unconstrained facial images. We also compared our

method with the SFS approach of [8] on more general unconstrained facial images. Since there are no ground truth shapes for these images, we only compared them visually. For reliable comparison, we directly ran our algorithm on the example images provided in [8]. Fig. 12 presents the comparison results, showing both the reconstructed face geometry and its textured display. We can see that our approach produced more accurate reconstruction of the overall shape, and recovered more geometrical details such as wrinkles and teeth. Although both methods perform SFS reconstruction, there is major difference on how the shape and illumination priors are derived. In [8] a reference face model is utilized as the shape prior to estimate illumination and initialize photometric normals; as the reference face model is not adapted to the target face shape, this can lead to unsatisfactory results. In comparison, with our method the

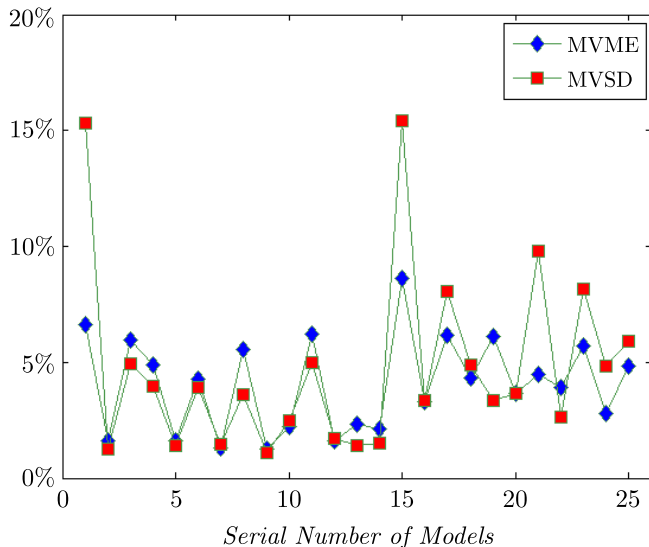


Figure 11: The MVME and MVSD for the reconstruction of 25 subjects from the Bosphorus database, from images with frontal pose and different expressions (happy, surprise, disgust).

medium face model is optimized to provide reliable estimates of the target shape and illumination, which enables more accurate reconstruction.

VIII. CONCLUSION

In this paper, we present a coarse-to-fine method to reconstruct a high-quality 3D face model from a single image. Our approach uses a bilinear face model and global corrective deformation fields to obtain a reliable initial face shape with large- and medium-scale features, which enables robust shape-from-shading reconstruction of fine facial details. The experiments demonstrate that our method can accurately reconstruct 3D face models from images with different poses and expressions, and recover the fine-scale geometrical details such as wrinkles and teeth. Our approach combines the benefits of low-dimensional face models and shape-from-shading, enabling more accurate and robust reconstruction.

REFERENCES

- [1] G. Stylianou and A. Lanitis, "Image based 3D face reconstruction: A survey," *International Journal of Image and Graphics*, vol. 9, no. 2, pp. 217–250, 2009.
- [2] V. Blanz and T. Vetter, "Face recognition based on fitting a 3d morphable model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1063–1074, 2003.
- [3] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pp. 787–796.
- [4] C. Cao, Q. Hou, and K. Zhou, "Displaced dynamic expression regression for real-time facial tracking and animation," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 43:1–43:10, 2014.
- [5] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2face: Real-time face capture and reenactment of rgb videos," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [6] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1999, Los Angeles, CA, USA, August 8-13, 1999*, 1999, pp. 187–194.
- [7] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 8, pp. 690–706, 1999.
- [8] I. Kemelmacher-Shlizerman and R. Basri, "3d face reconstruction from a single image using a single reference face shape," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 394–405, 2011.
- [9] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 3, pp. 413–425, 2014.
- [10] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *Journal of the Optical Society of America A*, vol. 4, no. 3, pp. 519–524, 1987.
- [11] M. Meytlis and L. Sirovich, "On the dimensionality of face space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1262–1267, 2007.
- [12] V. Blanz, K. Scherbaum, T. Vetter, and H.-P. Seidel, "Exchanging faces in images," *Computer Graphics Forum*, vol. 23, no. 3, pp. 669–676, 2004.
- [13] K. Scherbaum, T. Ritschel, M. Hullin, T. Thormhlen, V. Blanz, and H.-P. Seidel, "Computer-suggested facial makeup," *Computer Graphics Forum*, vol. 30, no. 2, pp. 485–492, 2011.
- [14] D. Vlasic, M. Brand, H. Pfister, and J. Popovic, "Face transfer with multilinear models," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 426–433, 2005.
- [15] J. P. Lewis, K. Anjyo, T. Rhee, M. Zhang, F. Pighin, and Z. Deng, "Practice and theory of blendshape facial models," in *Eurographics 2014 - State of the Art Reports*, S. Lefebvre and M. Spagnuolo, Eds. The Eurographics Association, 2014.
- [16] T. Weise, H. Li, L. Van Gool, and M. Pauly, "Face/off: Live facial puppetry," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2009, pp. 7–16.
- [17] T. Weise, S. Bouaziz, H. Li, and M. Pauly, "Realtime performance-based facial animation," *ACM Trans. Graph.*, vol. 30, no. 4, p. 77, 2011.
- [18] S. Bouaziz, Y. Wang, and M. Pauly, "Online modeling for realtime facial animation," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 40:1–40:10, 2013.
- [19] H. Li, J. Yu, Y. Ye, and C. Bregler, "Realtime facial animation with on-the-fly correctives," *ACM Transactions on Graphics (Proceedings SIGGRAPH 2013)*, vol. 32, no. 4, July 2013.
- [20] P. Hsieh, C. Ma, J. Yu, and H. Li, "Unconstrained realtime facial performance capture," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015, pp. 1675–1683.
- [21] C. Cao, Y. Weng, S. Lin, and K. Zhou, "3d shape regression for real-time facial animation," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 41:1–41:10, 2013.
- [22] C. Cao, D. Bradley, K. Zhou, and T. Beeler, "Real-time high-fidelity facial performance capture," *ACM Trans. Graph.*, vol. 34, no. 4, p. 46, 2015.
- [23] R. Gross, "Face databases," in *Handbook of Face Recognition*. Springer New York, 2005, pp. 301–327.
- [24] J.-D. Durou, M. Falcone, and M. Sagona, "Numerical methods for shape-from-shading: A new survey with benchmarks," *Computer Vision and Image Understanding*, vol. 109, no. 1, pp. 22 – 43, 2008.
- [25] E. Prados and O. Faugeras, "Shape from shading," in *Handbook of Mathematical Models in Computer Vision*, N. Paragios, Y. Chen, and O. Faugeras, Eds. Springer US, 2006, pp. 375–388.
- [26] I. Shimshoni, Y. Moses, and M. Lindenbaum, "Shape reconstruction of 3d bilaterally symmetric surfaces," *International Journal of Computer Vision*, vol. 39, no. 2, pp. 97–110, 2000.
- [27] W. Y. Zhao and R. Chellappa, "Illumination-insensitive face recognition using symmetric shape-from-shading," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000*, vol. 1, 2000, pp. 286–293 vol.1.
- [28] —, "Symmetric shape-from-shading using self-ratio image," *International Journal of Computer Vision*, vol. 45, no. 1, pp. 55–75, 2001.
- [29] J. J. Atick, P. A. Griffin, and A. N. Redlich, "Statistical approach to shape from shading: Reconstruction of three-dimensional face surfaces from single two-dimensional images," *Neural Computation*, vol. 8, pp. 1321–1340, 1996.
- [30] R. Dovgand and R. Basri, "Statistical symmetric shape from shading for 3d structure recovery of faces," in *Computer Vision - ECCV 2004*, T. Pajdla and J. Matas, Eds. Springer Berlin Heidelberg, 2004, pp. 99–113.
- [31] C. Wu, B. Wilburn, Y. Matsushita, and C. Theobalt, "High-quality shape from multi-view stereo and shading under general illumination," in *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, 2011*, pp. 969–976.



Figure 12: Face reconstructions from unconstrained images, using our method and the method from [8].

- [32] A. E. Ichim, S. Bouaziz, and M. Pauly, "Dynamic 3d avatar creation from hand-held video input." *ACM Trans. Graph.*, vol. 34, no. 4, p. 45, 2015.
- [33] I. Kemelmacher-Shlizerman and S. M. Seitz, "Face reconstruction in the wild," in *Proceedings of the 2011 International Conference on Computer Vision*, ser. ICCV '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 1746–1753.
- [34] J. Roth, Y. Tong, and X. Liu, "Unconstrained 3d face reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015, pp. 2606–2615.
- [35] —, "Adaptive 3d face reconstruction from unconstrained photo collections," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4197–4206.
- [36] L. De Lathauwer, *Signal processing based on multilinear algebra*. Katholieke Universiteit Leuven, 1997.
- [37] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun, "Joint cascade face detection and alignment," in *ECCV*, 2014, pp. 109–122.
- [38] C. Loop, "Smooth subdivision surfaces based on triangles," Master's thesis, The University of Utah, 1987.
- [39] D. Frolova, D. Simakov, and R. Basri, "Accuracy of spherical harmonic approximations for images of lambertian objects under far and near lighting," in *Computer Vision-ECCV 2004*. Springer, 2004, pp. 574–587.
- [40] H. Zhang, O. Van Kaick, and R. Dyer, "Spectral mesh processing," *Computer Graphics Forum*, vol. 29, no. 6, pp. 1865–1894, 2010.
- [41] K. Madsen, H. B. Nielsen, and O. Tingleff, "Methods for non-linear least squares problems," Informatics and Mathematical Modelling, Technical University of Denmark, 2004, 2nd edition.
- [42] A. Savran, N. Alyüz, H. Dibeklioglu, O. Çeliktutan, B. Gökberk, B. Sankur, and L. Akarun, "Bosphorus database for 3d face analysis," in *European Workshop on Biometrics and Identity Management*. Springer, 2008, pp. 47–56.
- [43] S. Agarwal, K. Mierle, and Others, "Ceres solver," <http://ceres-solver.org>.
- [44] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2009, pp. 296–301.